METHODS FOR COMPUTING BARYCENTRIC COORDINATES

GENERALIZED TO IRREGULAR N-GONS AND APPLICATIONS OF THE

SAME

5

INVENTORS

MATHIEU DESBRUN

ALAN H. BARR

MARK MEYER

10   This application claims the benefit of U.S. Provisional Application No. 60/267,804, filed February 8, 2001, which is hereby fully incorporated by reference herein as though set forth in full.

1.   Field of the Invention.

This invention relates generally to the field of barycentric coordinates for

15   representing interior points of polygons in terms of their boundary vertices, and, more specifically, methods of computing such coordinates, and applications of such coordinates.

2.   Related Art.

Barycentric coefficients are useful for representing interior points of a polygon

20   in terms of the polygon vertices. Barycentric coordinates are weights which, when applied to the vertices of a polygon, and the results added, yield the interior point.

Methods for computing triangular barycentric coordinates have been known for centuries, but efforts to generalize these methods to n-sided polygons have not been entirely successful, in part because the formulations resulting from these efforts

25   are subject to one or more weaknesses which limit their effectiveness and applicability.

For example, in Eugene L. Wachspress, "A Rational Finite Element Basis", Mathematics in Science and Engineering, Volume 114, Academic Press, 1975, a

method for computing barycentric coordinates for n-sided polygons is disclosed which can be explained with reference to Figure 1.

Referring to Figure 1, n-sided polygon 100 is illustrated, and it is desired to compute the barycentric coordinates for the interior point p, identified in the figure with numeral 102, in terms of the vertices of polygon 100. As is known, for an n-sided polygon, there will be n barycentric coordinates for an interior point, one corresponding to each of the n vertices. These barycentric coordinates, which can be referred to as α_j, $\forall j$, $j \in [1...n]$, are such that the following relationship holds:

$$\mathbf{p} = \sum_{j \in [1,...,n]} \alpha_j \mathbf{q_j} \tag{1}$$

where α_j refers to the barycentric coordinate applied to the vertex $\mathbf{q_j}$. (As used herein, a bolded item refers to a vector, while an unbolded item refers to a scalar).

According to the technique proposed by Wachpress, a weight w_j for vertex q_j is derived from the product of the area of the triangle 106 (shaded in Figure 1) formed by the three adjacent vertices $\mathbf{q_{j+1}}$, $\mathbf{q_j}$, and $\mathbf{q_{j-1}}$ and the areas of the n-2 interior triangles, identified with numerals 108, 110, and 112, formed by the point $\mathbf{p}$ and the polygon's adjacent vertices (but excluding the two interior triangles that contain the vertex $\mathbf{q_j}$). Mathematically, this formulation may be expressed as follows:

$$w_j = A(\mathbf{q_{j-1}}, \mathbf{q_j}, \mathbf{q_{j+1}}) \bullet \prod_{k \notin \{j, j+1\}} A(\mathbf{q_{k-1}}, \mathbf{q_k}, \mathbf{p}) \tag{2}$$

where the function A refers to the area of the triangle whose vertices are the arguments of the function.

This formulation is then repeated for each of the vertices in the polygon. The resulting weights may then be normalized in accordance with the following expression to form the barycentric coordinates for the point $\mathbf{p}$:

$$\alpha_j = w_j \Big/ \left(\sum_k w_k \right) \tag{3}$$

This formulation, however, is complex since the computation of the barycentric coordinate corresponding to a particular vertex depends on all the vertices of the polygon, and entails repetitive computations of triangle areas. Consequently,

the formulation is computationally intensive and therefore unacceptable for applications where a faster approach is needed.

SUMMARY

In a first embodiment, the invention provides a method for deriving barycentric coordinates for a point $\mathbf{p}$ within an n-sided polygon. For a particular coordinate w_j, corresponding to the vertex $\mathbf{q_j}$, this method embodies a formula which depends only on the edge $\mathbf{pq_j}$, and the two adjacent angles δ_j and γ_j. This formula may be expressed as follows:

$$w_j = \frac{\cot(\gamma_j) + \cot(\delta_j)}{\left\| \mathbf{p} - \mathbf{q_j} \right\|^2} \qquad (4)$$

This computation may be repeated for each of the n polygon vertices to result in barycentric coordinates corresponding to each of the n vertices. If necessary, the coordinates may then be normalized using expression (3).

In one implementation, for accuracy reasons, the cotangent function used in this formula is not computed through a trigonometric function call. Instead, in this implementation, the cotangent is derived from a division between the dot product of two vectors derived from the triangle involved, and the cross product of these two vectors. In particular, for a triangle having vertices $\mathbf{abc}$, the cotangent of the angle at vertex $\mathbf{b}$ may be derived using the following formulation:

$$\frac{\mathbf{bc} \bullet \mathbf{ba}}{\left\| \mathbf{bc} \times \mathbf{ba} \right\|} \qquad (5)$$

where $\mathbf{ba}$ is the vector $\mathbf{a} - \mathbf{b}$, and $\mathbf{bc}$ is the vector $\mathbf{c} - \mathbf{b}$.

In a second embodiment, the invention provides a method for deriving weights $\hat{w}_{ij}$ for expressing a vertex i in a mesh representation of an object surface in terms of its one-ring neighbors $j \in N(i)$. For a particular vertex $\mathbf{q_j}$, this method embodies a formula which depends only on the edge $\mathbf{q_i q_j}$, and the two adjacent angles δ_j and γ_j. This formula may be expressed as follows:

$$w_{ij} = \frac{\cot(\gamma_j) + \cot(\delta_j)}{\left\| \mathbf{q_i} - \mathbf{q_j} \right\|^2} \qquad (6)$$

This computation may be repeated for each of the vertices in the mesh representation to result in weights allowing each of the vertices to be expressed in terms of its one-ring neighbors. Normalized weights may then be derived from non-normalized weights using an expression analogous to expression (3).

In a third embodiment, the invention provides a method of parameterizing a mesh representation of an object surface using the formulation expressed in (6). This method begins with the step of computing the weights w_{ij} in 3D space (in contrast to parameter space) for each of the vertices in the mesh representation. For a vertex i, $i \in [1...n]$, the weights w_{ij} allow the vertex i to be expressed in terms of its one-ring neighbors $j \in N(i)$.

Once the weights have been derived, the method proceeds to fixing the positions of boundary vertices in parameter space, such as by assigning each of these vertices a position on a fixed boundary $\mathbf{C}$. For a boundary vertex i, the position assigned to this vertex in parameter space may be referred to as $\mathbf{C}_{u_i}$.

Then, the method proceeds to solving the following system of linear equations in order to derive the parameterization of the mesh representation:

$$\forall i, i \in [1....n], \begin{cases} \sum_{j \in N(i)} w_{ij}(\mathbf{u_i} - \mathbf{u_j}) = 0 & \textit{if i is an interior vertex} \\ \mathbf{u_i} = \mathbf{C_{u_i}} & \textit{if i is a boundary vertex} \end{cases} \quad (7)$$

where $\mathbf{u_i}$ is the vertex i in parameter space (and $\mathbf{u_j}$ is the vertex j in parameter space), and $\mathbf{C_{u_i}}$ is the boundary position in parameter space assigned to the boundary vertex i.

The result of solving this system of linear equations is the coordinates $\mathbf{u_i}$ of each of the vertices i in parameter space, $i \in [1...n]$.

Other systems, methods, features and advantages of the invention will be or will become apparent to one with skill in the art upon examination of the following figures and detailed description. It is intended that all such additional systems, methods, features and advantages be included within this description, be within the scope of the invention, and be protected by the accompanying claims.

BRIEF DESCRIPTION OF THE DRAWINGS

The components in the figures are not necessarily to scale, emphasis instead being placed upon illustrating the principles of the invention. In the figures, like reference numerals designate corresponding parts throughout the different views.

Figure 1 illustrates a representation of an n-sided polygon partitioned into triangles useful for computing the barycentric coordinates of a point p using the prior art formula of Wachpress.

Figure 2 illustrates a representation of an n-sided polygon showing the edge pq_j and the adjacent angles δ_j and γ_j useful for computing the barycentric coordinates of a point p using a method according to the invention.

Figure 3 illustrates parameterization of a mesh representation of an object surface.

Figure 4 illustrates application of a method according to the invention to computing weights for expressing a vertex of a mesh representation in terms of its one-ring neighbors.

Figure 5 illustrates the vectors and triangle vertices which are useful in computing the cotangent of one of the vertex angles according to a method which avoids a trigonometric function call.

Figure 6 is a simplified flowchart illustrating one embodiment of a method of parameterizing a mesh representation of an object surface using a formula according to the invention.

DETAILED DESCRIPTION

As utilized herein, terms such as "about" and "substantially" are intended to allow some leeway in mathematical exactness to account for tolerances that are acceptable in the trade, e.g., any deviation upward or downward from the value modified by "about" or "substantially" by any value in the range(s) from 1% to 20% of such value.

Referring to Figure 2, in a first embodiment of the invention, a method for deriving non-normalized barycentric coordinates for a point $\mathbf{p}$ within an n-sided polygon is provided. For a particular coordinate w_j, corresponding to the vertex $\mathbf{q_j}$, identified with numeral 202, this method embodies a formula which depends only on the edge $\mathbf{pq_j}$, identified with numeral 204, and the two adjacent angles δ_j and γ_j, identified respectively with numerals 206 and 208. This formula may be expressed as follows:

$$w_j = \frac{\cot(\gamma_j) + \cot(\delta_j)}{\|\mathbf{p} - \mathbf{q_j}\|^2} \qquad (8)$$

This computation may be repeated for each of the n polygon vertices to result in non-normalized barycentric coordinates corresponding to each of the n vertices. Normalized coordinates may then be derived from the non-normalized coordinates using the following expression:

$$\alpha_j = \frac{w_j}{\left(\sum_k w_k\right)} \qquad (9)$$

Compared to the Wachpress formulation (2), this formulation has the advantage that it is local and hence computationally simple: only the edge $\mathbf{pq_j}$ and its two adjacent angles δ_j and γ_j are needed.

The coordinates derived from this formula possess several desirable properties. First, they satisfy the following affine combination property:

$$\mathbf{p} = \sum_{j \in [1..n]} \alpha_j \mathbf{q_j}, \qquad with \ \sum_{j \in [1 \ n]} \alpha_j = 1 \qquad (10)$$

This property may be expressed in the following form using the non-normalized coordinates w_i:

$$\sum_{j \in [1.. n]} w_j (\mathbf{q_j} - \mathbf{p}) = 0 \qquad (11)$$

Second, the coordinates $\{\alpha_j\}_{j=1 \ n}$ (and the normalized coordinates) are smoothly dependent on $\mathbf{p}$ and the vertices of the polygon. This ensures smoothness in the variation of the position of $\mathbf{p}$ responsive to any movement in the vertex $\mathbf{q_j}$.

Third, if the polygon is convex, the coordinates $\{\alpha_j\}_{j=1 \ldots n}$ satisfy the following convex combination property:

$$\alpha_j \geq 0 \ \forall j \in [1 \ldots n] \tag{12}$$

This guarantees no under- and over-shooting in the coordinates: all the coordinates will be between zero and one.

In one implementation, for accuracy reasons, the cotangent function used in the formula (7) is not computed through a trigonometric function call. Instead, in this implementation, the cotangent is derived from a division between the dot product of two vectors derived from the triangle involved, and the cross product of these two vectors. In particular, referring to Figure 3, for a triangle having vertices **abc**, the cotangent of the angle θ at vertex **b**, which angle is identified with numeral 302, may be derived using the following formulation:

$$\frac{\mathbf{bc} \bullet \mathbf{ba}}{\|\mathbf{bc} \times \mathbf{ba}\|} \tag{13}$$

where **ba** is the vector **a** − **b**, and **bc** is the vector **c** − **b**.

In one example, the formula (8) may be implemented using the following pseudo-code:

```
// Compute the barycentric weights for a point p in an n-gon Q
// Assumes p is strictly within Q and the vertices qj are ordered.
computeBarycentric(vector2d p, polygon Q, int n, real w[ ])
        weightSum = 0
        foreach vertex qj of Q:
                prev = (j + n − 1) mod n
                next = (j + 1) mod n
                wj = (cotangent(p, qj, qprev) + (cotangent(p, qj, qnext)/‖p − qj‖²
                weightSum += wj
// Normalize the weights
foreach weight wj:
        wj /= weightSum


// Compute the cotangent of the non-degenerate triangle abc at vertex b
cotangent(vector2d a, vector2d b, vector2d c)
        vector2d ba = a − b
        vector2d bc = c − b
```

7

$$\text{return}\left(\frac{\mathbf{bc} \bullet \mathbf{ba}}{\|\mathbf{bc} \times \mathbf{ba}\|}\right)$$

The formula (8), any derivation thereof, and any instructions or program code implementing the same, may be tangibly embodied, through storage or otherwise, on or in a processor readable medium or memory including but not limited to RAM, ROM, PROM, EPROM, EEPROM, hard disk, floppy disk, CD-ROM, DVD, flash memory, etc. For purposes of this disclosure, the term "memory" refers to any medium which is capable of storing data.

Referring to Figure 4, in a second embodiment of the invention, a method is provided for deriving weights w_{ij} for expressing a vertex $\mathbf{q_i}$ in a mesh representation 400 of an object surface in terms of its one-ring neighbors $\mathbf{q_j}$, $\forall j \in N(i)$. For a particular vertex $\mathbf{q_j}$, identified with numeral 402, and a neighbor vertex $\mathbf{q_j}$, identified with numeral 402, this method embodies a formula which depends only on the edge $\mathbf{q_i q_j}$, identified with numeral 404, and the two adjacent angles δ_j and γ_j, identified respectively with numerals 406 and 408. This formula may be expressed as follows:

$$w_{ij} = \frac{\cot(\gamma_j) + \cot(\delta_j)}{\|\mathbf{q_i} - \mathbf{q_j}\|^2} \qquad (14)$$

This computation may be repeated for each of the one-ring neighbors of $\mathbf{q_i}$ resulting in weights allowing the vertex $\mathbf{q_i}$ to be expressed in terms of its one-ring neighbors. Normalized weights may then be derived from non-normalized weights using an expression analogous to expression (9).

This series of computations may be repeated for each of the other vertices in the mesh representation, thus allowing each vertex to be expressed in terms of its one-ring neighbors.

Thus, in contrast to the previous embodiment which concerned expressing an interior point of an n-sided polygon in terms of its vertices, in this embodiment, a vertix of an n-sided polygon may be expressed in terms of its one-ring neighbors, which may be vertices of the same or a different polygon.

The formula (14), any derivation thereof, and any instructions or program code implementing the same, may be tangibly embodied, through storage or otherwise, on or in a processor readable medium or memory including but not limited to RAM, ROM, PROM, EPROM, EEPROM, hard disk, floppy disk, CD-ROM, DVD, flash memory, etc.

In a third embodiment, the invention provides a method of parameterizing a mesh representation of an object surface using the formula (14). Referring to Figure 5, parameterization is a transformation of a mesh representation 500 from 3D (x, y, z) space to parameter (u, v) space, and is performed to ease data manipulation and to achieve a better sampling of the original object surface than is achieved through the 3D mesh representation. The mesh representation, after being mapped into the parameter space, is identified with numeral 502.

Figure 6 illustrates a flowchart of the method. Referring to Figure 6, the method begins with step 602, which comprises computing the weights w_{ij} in 3D space (in contrast to parameter space) for each of the vertices in the mesh representation. For a vertex q_i, $i \in [1...n]$, the weights w_{ij} are derived through application of the formula (14).

Once the weights have been derived, the method proceeds to step 604, which comprises fixing the positions of boundary vertices in parameter space, such as by assigning each of these vertices a position on a fixed boundary C. For a boundary vertex i, the position assigned to this vertex in parameter space may be referred to as C_{u_i}. Thus, in the example illustrated in Figure 5, the contour C is fixed in parameter space, and each of the boundary vertices of the mesh representation 500 is assigned a position on the contour C. For example, the vertex 504 may be assigned the position 508 on contour C, and the vertex 506 may be assigned the position 510 on contour C.

Then, the method proceeds to step 606, which comprises solving the following system of linear equations in order to derive the parameterization of the mesh representation:

$$\forall i, i \in [1....n], \begin{cases} \sum_{j \in N(i)} w_{ij} (\mathbf{u_i} - \mathbf{u_j}) = 0 & \textit{if i is an interior vertex} \\ \mathbf{u_i} = \mathbf{C}_{\mathbf{u}_i} & \textit{if i is a boundary vertex} \end{cases} \quad (15)$$

where $\mathbf{u_i}$ is the vertex i in parameter space (and $\mathbf{u_j}$ is the vertex j in parameter space), and $\mathbf{C}_{\mathbf{u}_i}$ is the boundary position in parameter space assigned to the boundary vertex i.

The result of solving this system of linear equations is the coordinates $\mathbf{u_i}$ of each of the vertices in parameter space, $i \in [1...n]$. In the example of Figure 5, this step yields the coordinates of the vertex 512 in parameter space (which coordinates are identified with numeral 514).

For additional information on parameterization, and the application of formula (14) to parameterization, the reader is referred to Mathieu Desbrun et al., "Intrinsic Parameterization of Surface Meshes," pp. 1-8, submitted for publication in EUROGRAPHICS 2002, and hereby fully incorporated herein by reference as through set forth in full. See also Mathieu Desbrun et al., "Generalizing Barycentric Coordinates to Irregular N-gons," pp. 1-5, submitted for publication in EUROGRAPHICS 2002, and hereby fully incorporated herein by reference as through set forth in full.

The foregoing method may be tangibly embodied, through storage of instructions, program code, or the like implementing the method or otherwise, on or in a processor readable medium or memory including but not limited to RAM, ROM, PROM, EPROM, EEPROM, hard disk, floppy disk, CD-ROM, DVD, flash memory, etc.

While various embodiments of the invention have been described, it will be apparent to those of ordinary skill in the art that many more embodiments and implementations are possible that are within the scope of this invention.